MULTIGRID FOR THE HELMHOLTZ EQUATION: CHOICE OF THE COARSE SCALE OPERATOR

CHRISTIAAN C. STOLK, MOSTAK AHMED, AND SAMIR KUMAR BHOWMIK

Abstract. We study the convergence of multigrid schemes for the Helmholtz equation, focusing in particular on the choice of the *coarse scale operators*. Assuming that the coarse scale solutions should approximate the true solutions, the coarsest level grid must have a minimum number, say G_c, of points per wavelength. The oscillating nature of the solutions implies the requirement $G_c > 2$. However, in examples the requirement can be more like $G_c \gtrsim 8$, in a trade-off involving also the amount of damping present and the number of multigrid iterations. We conjecture that the main factor affecting G_c is the difference in phase speeds between the coarse and fine scale operators. Standard 5-point finite differences in 2-D are our first example. A new coarse scale 9-point operator is constructed to match the fine scale phase speeds. We then compare phase speeds and multigrid performance of standard schemes, with a new scheme that uses the new coarse scale operators and a modified smoother. The required G_c is reduced from about 8 to about 4, while at the same time less damping is needed so that waves propagate over $\sim$ 100 wavelengths in the new scheme. Next we consider extensions of the method to more general cases. In 3-D comparable results are obtained with standard 7-point differences and optimized 15-point coarse grid operators, leading to an order of magnitude reduction in the number of unknowns for the coarsest scale linear system. Finally we show how to include PML boundary layers, using a regular grid finite element method. Matching coare scale operators can easily be constructed for other discretizations. The method is therefore potentially useful for a large class of discretized high-frequency Helmholtz equations.

1. Introduction

The high-frequency Helmholtz equation forms a major challenge in numerical analysis. When the domain is large compared to the wavelength, i.e. large k, the large indefinite systems resulting from the discretization are difficult to solve. High-frequency Helmholtz problems have applications in various simulation problems and in inverse problems, e.g. in exploration seismology and acoustic scattering.

Multigrid methods have been used for Helmholtz equations, but generally not in the conventional setup as used in elliptic problems (see e.g. [16]). The reason is that conventional multigrid methods perform poorly. Issues are the divergence of the iterative smoothing methods and the resolution of the coarse grid. To be precise, denote by G_c the minimum number of grid points per wavelength in the coarsest grid. If the coarse problem solutions are to approximate the solutions of the original problem, the oscillatory nature of the solutions leads to the requirement $G_c > 2$. However, in examples the requirement is more like $G_c \gtrsim 8$.

For large k problems the coarsest scale problem can still be quite large, leading to large memory requirements. Shifted Laplacian methods [7] provide a way to address this. They use a multigrid cycle as a preconditioner for an outer iterative method like BiCGSTAB. However, the multigrid is not applied to the original operator, but to a modified operator with a strong damping term added. Coarse grids coarser than the wavelength scale are used ($G_c < 1$) (see also [5]). This leads to a method that is convergent and with relatively low memory use and cost per iteration, but with quite large iteration numbers, increasing with the domain size.

Direct methods on the other hand also remain relevant, in particular in view of recent results on double sweeping domain decomposition methods, like the moving PML method of [6] or the double sweeping method of [14]. The double sweeping domain decomposition methods appear to have the best, near-linear scaling for the cost per solve, but have the disadvantage of a large memory use. See [12] for a parallel implementation of the method of [6]. Other methods include those in [2, 3, 17]. In all cases it is important to distinguish the behavior in non-resonant cases vs. resonant cases. In variable coefficient media with resonances the performance of the iterative methods discussed above tends to deteriorate strongly.

In this paper we study multigrid methods from the following point of view. We remain in the regime $G_c > 2$, aiming at fast convergence, say $\lesssim 20$ iterations for reduction of the residual by 10^{-6}. However, we challenge the conventional choices made in the design of multigrid methods. With new methods we obtain a reduction of G_c from about 8 to about 4. As examples we study the standard 5-point and 7-point stencils (in 2-D and 3-D resp.) for the original, fine scale operator, and a regular grid finite element method for which we show how to include PML boundary layers.

To obtain these results we briefly analyze and modify the smoothing steps in the algorithm, and then focus on the choice of the coarse grid operators. Standard choices for coarse scale operators are the Galerkin approximation, or to use the same discretization scheme as the fine scale operator. We will show that these are suboptimal by constructing optimized coarse scale operators that perform much better, yielding the aforementioned factor 2 improvement of G_c in combination with substantially less damping required in the equation. Note that a factor 2 improvement in G_c reduces the number of unknowns for the coarsest scale direct solver by 8 in 3-D, and the computational cost and memory use by an even higher factor. Moreover, there is nothing special about the discretizations studied in this paper, and we conjecture that in fact our method is applicable to any regular grid fine scale discretization.

We now introduce the setup in more detail. The Helmholtz equation reads

$$Lu \stackrel{\text{def}}{=} -\Delta u - ((1+\alpha i)k(x))^2 u = f, \quad \text{in } \Omega \subset \mathbb{R}^n \tag{1}$$

where Δ is the Laplacian and Ω is a rectangular block. We assume Dirichlet boundary conditions the boundary $\partial\Omega$. PML layers will be present in some of the examples, they will be discussed in section 5.2. The parameter k in general depends on $x \in \Omega$. Here α is a parameter for the damping, that will mostly be independent of x[1]. The corresponding undamped operator will be denoted by

$$H = -\Delta - k^2. \tag{2}$$

The imaginary contribution $i\alpha k$ leads to exponential decaying solutions. For example, in 1-D for constant k there are solutions

$$e^{ikx - \alpha k x}. \tag{3}$$

This limits the propagation distance of the waves. This distance, measured by the number of wavelengths for the amplitude to be reduced by a factor 10, is given by

$$D(\alpha) = \frac{\log(10)}{2\pi\alpha}. \tag{4}$$

Values of α will range from $1.25 \cdot 10^{-3}$ to 0.02, small enough for applications. In presence of PML layers α will be set to zero. The experiments will show that, for the optimized coarse grid methods good convergence starts roughly around 0.0025 or at $D(\alpha) \approx 145$ wave lengths assuming around 4 points per wavelength in the coarsest grid.

[1]Other authors sometimes use $-\Delta - (1+i\alpha)k^2$, i.e. with the factor $(1+i\alpha)$ outside the square. The sign of $i\alpha$ is related our choice of temporal Fourier transform $f(t) = e^{-i\omega t}\hat{f}(\omega)$.

Multigrid methods consist of several components. For example, a two-grid method consists of the following steps: presmoothing using an iterative smoothing method; restriction of the residual to the coarse grid; solving a coarse grid equation; prolongation of the solution to the fine grid; suppressing remaining errors using postsmoothing, using again an iterative smoothing method. In our method we introduce two main changes compared to conventional multigrid methods. The first concerns the smoothing, the second the coarse grid operator. In a two-grid method, this will lead to three discrete versions of the operator L: the original, fine scale operator L_h, the operator used for smoothing $L_{\mathrm{S},h}$ and a coarse scale operator $L_{\mathrm{c},2h}$.

As for the smoothing, it is well known that iterative smoothing methods are unstable for Helmholtz problems. We find that this can be solved by using instead the Laplacian operator in the smoother, or, when a PML layer is present, by using the positive definite Helmholtz operator $-\Delta + k^2$. The error caused by this modification is handled by the outer iterative method, in such a way that the overall effect of this modification is positive.

The second component to be modified is the coarse grid operator. Our claim is that the coarse problem should have the same *phase speed* or numerical dispersion as the fine scale problem. Indeed over large distances the phase errors lead to large differences between the approximate, coarse grid solution and the true solution. Alternatively, from Fourier analysis of multigrid methods one can argue that the *inverse of the symbols* for the coarse scale and original operators should match, which in turn implies that the zero-sets and hence the phase speeds should match. To obtain coarse scale operators with these properties we optimize the coefficients in the finite difference scheme, minimizing the difference of the symbols, with a strong weighting around the zeroset of the fine scale symbol.

We first take the example of standard 5-point finite differences in 2-D. After computing the optimized coefficients, we compare the phase speeds associated with various finite difference discretizations, analyze the two-grid convergence factors for the conventional and modified methods, and perform numerical experiments. The multigrid convergence is good or poor, precisely when the phase speeds match well or poorly, respectively. Based on the two-grid convergence factors, this conclusion is derived in constant coefficient, 2-D case. Numerical experiments show that similar conclusions hold for the multigrid method, when the coefficient k is spatially varying. In addition we study in 2-D the optimized finite difference scheme by Jo, Shin and Suh [9]. Their scheme has accurate phase speeds for all values of G (the number of grid points per wavelength) for G down to 4, which means the phase speeds match well if this scheme is used at all levels of the multigrid method. It also performs well. The advantage of our own optimized scheme is that *it can be adapted for any regular grid discretization* of the original problem.

Quantitative results are as follows. We observe that for α in the order of 0.25 % we still have good convergence for 4 points per wave length with the optimized method. Regular multigrid with 8 points per wave length however, only performs well if $\alpha \gtrsim 2$ %. Note that the actual value of G_{c}, whether it is say 8 or more, or around 3.5 or 4, can make a great difference. Reducing the grid by a factor of 2 in each direction, leads in 3-D to a 2^3 fold reduction in the number of unknowns and an even larger reduction in the computational cost of the sparse factorization and the memory use, up to 2^6 for the sparse factorization and $2^{9/2}$ for the memory use [8].

Next we consider more general settings than 5-point finite differences. For the 3-D problem we construct optimized coarse scale operators for the standard 7-point finite difference scheme. Numerical tests show similar results as for the 2-D case, with good convergence for $G_{\mathrm{c}} \gtrsim 4$.

In practice problems on rectangular domains often occur in combination with absorbing boundary conditions or absorbing boundary layers, such as PML layers [1, 4]. These then provide damping in the equation, so that we set $\alpha = 0$. In numerical experiments, straightforward insertion of a PML layer was observed to lead to very poor convergence

or no convergence at all in the situations we are interested in ($G_c \gtrsim 4$). To address this we consider an adapted coarsening strategy. In the PML layer no coarsening takes place in the direction normal to the boundary. This is the direction in which a fast variation of the coefficient σ used in the PML takes place. Because this leads to (partly) irregular grids, it is natural to consider this in the context of finite elements. Optimized coarse grid operators for a first order rectangular finite element discretization are computed, and we show in 2-D examples that this again results in good convergence of the multigrid method for $G_c \gtrsim 4$, and propapagation distances of up to 200 wavelengths.

The setup of the paper is as follows. The next section focusses on the phase speeds. Here we construct the new, optimized finite difference operators for the various cases and compare the phase speed errors in the standard and the new coarse grid operators. Then in section 3 we describe the multigrid methods used in this study. In section 4 we present Fourier analysis of the two-grid methods. Section 5 contains the results of numerical simulations: First the comparison of the standard and the new methods for 2-D finite differences, and then the extensions to multigrid, to 3-D and to problems with PML boundary layers.

2. Phase speeds and optimized coarse scale operators

We first recall the notions of dispersion relation and phase speed for constant coefficient linear time dependent partial differential equations [15]. If P is such an operator, and

$$P(\xi,\omega) = e^{-i\xi\cdot x+i\omega t}(Pe^{i\xi\cdot x-i\omega t}), \tag{5}$$

denotes its symbol, then the dispersion relation is the set of (ξ,ω) where $P(\xi,\omega) = 0$. For the Helmholtz equation, the symbol is a function of the spatial wave number ξ, with parameter ω

$$H(\xi;\omega) = e^{-i\xi\cdot x}(He^{i\xi\cdot x}) \tag{6}$$

and the dispersion relation is understood to be the ω-dependent set

$$\{\xi \in S \,|\, H(\xi;\omega) = 0\}, \tag{7}$$

where for the continuous operator $S = \mathbb{R}^n$, and it is assumed that $\alpha = 0$. For ξ such that $H(\xi;\omega) = 0$, the number $\frac{\omega}{\|\xi\|}$ is called the phase speed $v_{\rm ph}$ associated with a plane wave solution. For the continuous operator $v_{\rm ph}$ is constant equal to c.

For finite difference discretizations of the Helmholtz operator the definitions (6) and (7) remain valid except that S is given by the fundamental domain $S = [-\pi/h, \pi/h]^n$. In the typical case that H is increasing along half lines from the origin, the phase speed is a function of angle. We will compute the dimensionless phase speed $\frac{v_{\rm ph}}{c}$ as a function of another dimensionless quantity, the number of points per wavelength G, or its inverse $1/G$ (G is related to the dimensionless quantity kh by $kh = \frac{2\pi}{G}$).

This discussion will mostly involve only two scales, a fine scale and a coarse scale with double the grid parameter. In multigrid with more than two levels, the additional levels are assumed to be increasingly fine, because the coarsest level is restricted by the wave length. In general we expect that the phase speed difference between the two coarsest levels is the most important, since for finer grids these differences automatically become smaller as the discretization becomes a better approximation of the true operator.

Next we first show the behavior of the phase speeds for some well known finite difference schemes and then construct a finite difference method that is optimized so that its dispersion relation matches that of the standard 5-point method. After this we consider the 7-point operator in 3-D and the finite element method used with PML layers in 2-D.

2.1. **Phase speeds of some well known finite difference schemes.** The standard 5-point finite difference discretization (fd5) of the 2-D Helmholtz operator is given by

$$(H_h^{\mathrm{fd5}}u)_{i,j} = h^{-2}(4u_{i,j} - u_{i-1,j} - u_{i+1,j} - u_{i,j-1} - u_{i,j+1}) - k^2 u_{i,j}. \tag{8}$$

Its symbol is easily shown to be

$$\sigma_h^{\mathrm{fd5}}(\xi) = h^{-2}(4 - 2\cos(h\xi_1) - 2\cos(h\xi_2)) - k^2, \tag{9}$$

where $\xi = (\xi_1, \xi_2)$ denotes the wave vector. The phase speed as a function of angle is easily computed using a root finding algorithm and shown in Figure 1(a). The phase speeds as a function of $|\xi|$ are given for 4 angles $0°$, $15°$, $30°$ and $45°$.

When such a scheme is used in a multigrid method, say a two-grid method for the purpose of this argument, standard choices for the coarse scale operator are to use the same discretization, or to use a Galerkin discretization. The comparison between the phase speeds of a coarse scale operator H_h^{fd5} and a fine scale operator $H_{h/2}^{\mathrm{fd5}}$ is given in Figure 1(b). Here G refers to the coarse scale. The Galerkin method, using "full weighting" restriction and prolongation operators [16] is easily shown to have symbol

$$\begin{aligned}\sigma_h^{\mathrm{gal}}(\xi) = {} & 3h^{-2} - \frac{9}{16}k^2 + (-h^{-2} - \frac{3}{16}k^2)(\cos(\xi_1 h) + \cos(\xi_2 h)) \\ & + (-\frac{1}{2}h^{-2} - \frac{1}{32}k^2)(\cos((\xi_1+\xi_2)h) + \cos((\xi_1-\xi_2)h)).\end{aligned} \tag{10}$$

The phase speeds and the phase speed difference between a coarse scale Galerkin and a fine scale fd5 method are shown in Figure 1(c) and (d). The phase speed differences are on the order of 0.01 or 0.02 for $G = 8$ and larger for G smaller.

We contrast this with the optimized finite difference method of Jo, Shin and Suh [9], sometimes called the mixed grid operator. This operator, here called JSS operator for brevity, is given by

$$\begin{aligned}(H_h^{\mathrm{jss}}u)_{i,j} = {} & ((2+2a)h^{-2} - ck^2)u_{i,j} + (-ah^{-2} - dk^2)(u_{i+1,j} + u_{i-1,j} + u_{i,j+1} + u_{i,j-1}) \\ & + (-\frac{1-a}{2}h^{-2} - \frac{(1-c-4d)}{4}k^2)(u_{i+1,j+1} + u_{i-1,j+1} + u_{i+1,j-1} + u_{i-1,j-1})\end{aligned} \tag{11}$$

where $a = 0.5461$, $c = 0.6248$ and $d = 0.9381 \cdot 10^{-1}$. The symbol of this operator hence equals

$$\begin{aligned}\sigma_h^{\mathrm{jss}}(\xi) = {} & (h^{-2}(2+2a) - k^2 c) + 2(-h^{-2}a - k^2 d)(\cos(\xi_1 h) + \cos(\xi_2 h)) \\ & + 2(-h^{-2}\frac{1-a}{2} - \frac{(1-c-4d)}{4}k^2)(\cos((\xi_1+\xi_2)h) + \cos((\xi_1-\xi_2)h)).\end{aligned} \tag{12}$$

The phase speed and relative phase speed difference between H_{2h}^{jss} and H_h^{jss} are given in Figure 1(e) and (f). The phase speed difference between coarse and fine level operators, both using the JSS method, is reduced to around $2.5 \cdot 10^{-3}$ for G down to 4, i.e. a very substantial improvement both in G and in the size of the phase speed difference.

The operators L_h^{fd5}, L_h^{gal} and L_h^{jss}, with $\alpha \neq 0$ are obtained simply by replacing k^2 by $((1+i\alpha)k)^2$.

2.2. **An optimized finite difference scheme.** We now construct a finite difference method with phase speed matching that of the standard 5-point method, with comparable accuracy as the JSS method, and even down to $G = 3$. The construction is easily generalized to 3-D and to other fine scale methods.

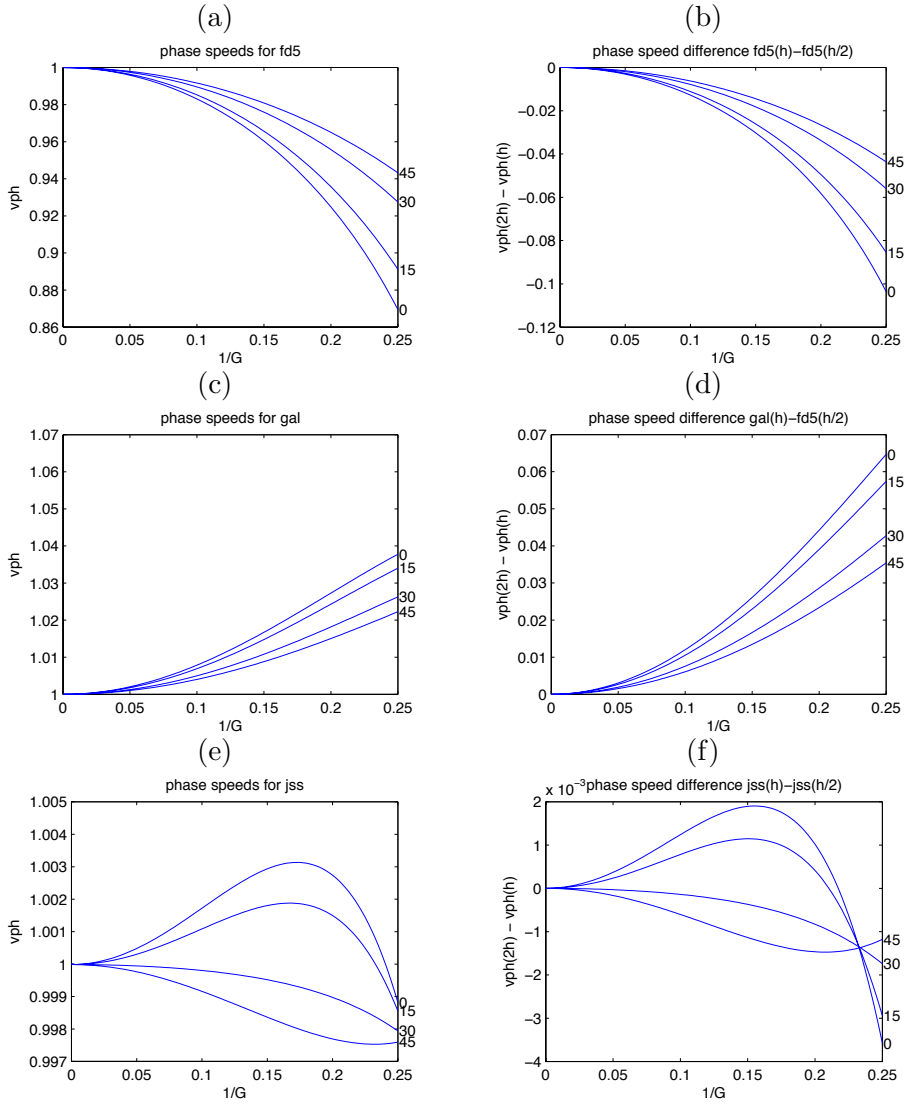


FIGURE 1. Dimensionless phase speed curves and difference between fine and coarse scale phase speeds; (a), (c) and (e): phase speed for the fd5, Galerkin and JSS method; (b), (d) and (f): phase speed differences $v_h^{\mathrm{fd5}} - v_{h/2}^{\mathrm{fd5}}$, $v_h^{\mathrm{gal}} - v_{h/2}^{\mathrm{fd5}}$, and $v_h^{\mathrm{JSS}} - v_{h/2}^{\mathrm{JSS}}$, with angles 0°, 15°, 30°, 45°

We consider finite difference operators of the following form

$$
\begin{aligned}
H_h^{\mathrm{opt}} u(x_{i,j}) = & -k^2 u_{i,j} + h^{-2}(a_1 u_{i,j} + a_2(u_{i-1,j} + u_{i+1,j} + u_{i,j-1} + u_{i,j+1}) \\
& + a_3(u_{i-1,j-1} + u_{i-1,j+1} + u_{i+1,j-1} + u_{i+1,j+1}) \\
& + a_4(u_{i-2,j} + u_{i+2,j} + u_{i,j-2} + u_{i,j+2})),
\end{aligned} \tag{13}
$$

where we include the possibility that $a_3 = 0$, or $a_4 = 0$. Partly based on the numerical results below, we will in fact take $a_4 = 0$ in most of work. The associated symbol is

$$
\begin{aligned}
\sigma_h^{\mathrm{opt}} = & -k^2 + h^{-2}(a_1 + 2a_2(\cos(h\xi_1) + \cos(h\xi_2)) \\
& + 2a_3(\cos(h(\xi_1+\xi_2)) + \cos(h(\xi_1-\xi_2)) + 2a_4(\cos(2h\xi_1) + \cos(2h\xi_2)).
\end{aligned} \tag{14}
$$

For the operator L_h^{opt} with $\alpha \neq 0$, the a_j are still a function of k, and the term $k^2 u_{i,j}$ is replaced by $((1+\alpha i)k)^2 u_{i,j}$.

The coefficients a_j, will be chosen as a function of $(h, h_{\rm f}, k)$, where $h = h_{\rm c}$ is the grid distance of the grid the operator is used in, i.e. the coarse grid, and $h_{\rm f}$ represents the grid distance of the fine scale operator to be approximated. However, from dimensional considerations, the coefficients a_j only have to depend on two parameters, namely $a_j = a_j(\frac{h_{\rm f}}{h}, hk)$, where the parameter hk can also be replaced by the parameter $1/G$ as noted above.

The optimization procedure implements two main features

- The optimized coarse symbol is an approximation to the fine symbol for all ξ.
- However, the accuracy of the approximation depends on the position in the ξ plane. In particular, the approximation should be highly accurate around the zero-set of the fine symbol in order to reproduce the phase speeds.

This is done in an ad hoc fashion, choosing a set of points P_j in the ξ plane, with for each point an associated weights $w_j > 0$ and subsequently finding (a_1, a_2, a_3, a_4), (or (a_1, a_2, a_3) in case $a_4 = 0$ or (a_1, a_2, a_4) in case $a_3 = 0$) that minimize

$$\sum_j w_j (\sigma^{\rm fine}(P_j) - \sigma^{\rm opt}(P_j))^2. \tag{15}$$

The starting point for the choice of the P_j is the closed curve $r = r_0(\theta)$ (in polar coordinates), such that

$$\sigma^{\rm fine}(r_0(\theta)\cos\theta, r_0(\theta)\sin\theta) = 0. \tag{16}$$

We choose n_θ points on the set $r = r(\theta)$, equally spaced in angle, with $w_j = 1$. Then we choose points at $0.85r(\theta)$ and at $1.15r(\theta)$, with weight $1/6$, and points at $0.5r(\theta)$ and at $1.6r(\theta)$ with weight 0.05. Finally we add points at $(0,0), (0,\pi), (\pi,\pi)$ with weights 0.01, 0.01, 0.01. These ad hoc choices are justified by the good behavior of the obtained phase speeds, as well as the obtained multigrid methods (see below). But other choices could be possible as well.

We computed results for 3 coefficients, a_1, a_2, a_3. For $h_{\rm f}/h = 0.5$ the result is given in Figure 2. We observe that the numerical optimization indeed produces a result and the coefficients appear to depend smoothly on hk.

Choosing values of $h_{\rm f}/h$ different from $1/2$ is useful in multigrid with more than two levels, see below. The maximum over θ and the root-mean-square over θ of the phase error are given in Figure 4, where $h_{\rm f}/h$ is given by $1/2, 1/4, 1/8$ respectively.

As can be seen from these figures, the phase speed differences are reduced to around $2.5 \cdot 10^{-3}$ for $G_{\rm c}$ downto to 3, comparable to JSS, and with in fact a slight improvement below $G_{\rm c} = 4$.

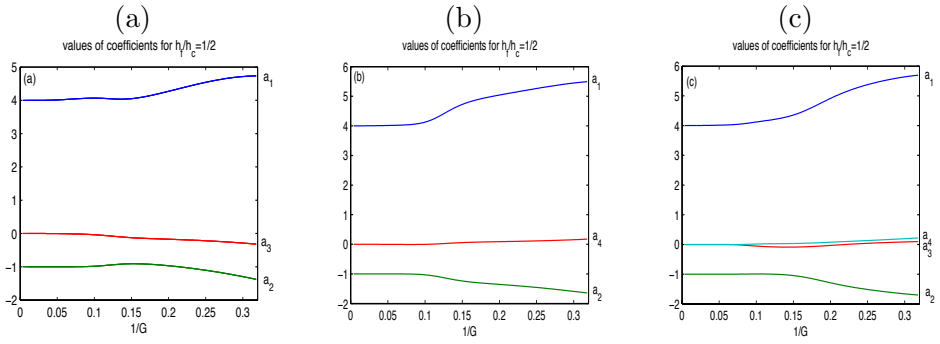


FIGURE 2. Coefficients a_j as function of $1/G = \frac{hk}{2\pi}$; (a) coefficients a_1, a_2, a_3 when $a_4 = 0$; (b) coefficients a_1, a_2, a_4 when $a_3 = 0$; (c) coefficients a_1, a_2, a_3, a_4.

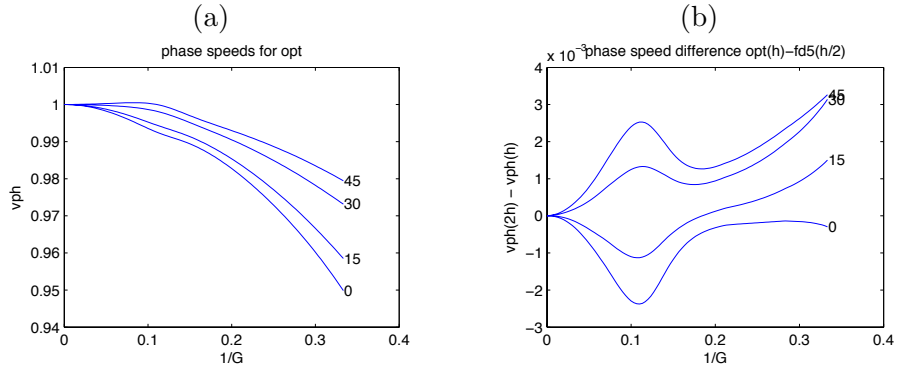


FIGURE 3. (a) Dimensionless phase speed for the opt method; (b) phase speed difference $v_h^{\mathrm{opt}} - v_{h/2}^{\mathrm{fd5}}$

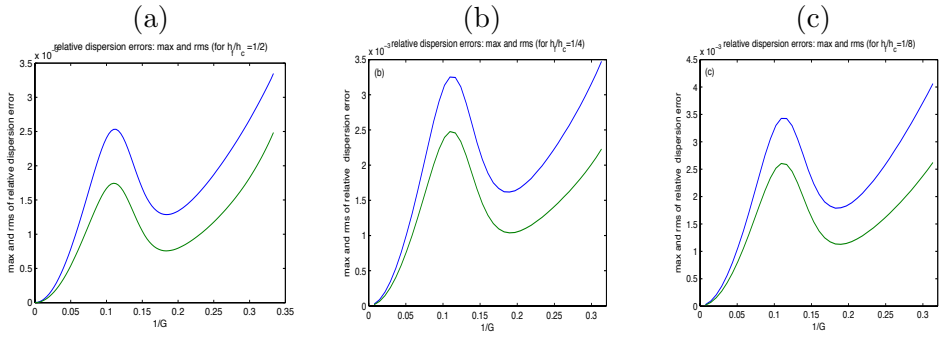


FIGURE 4. Phase speed error for the different values of h_{f}/h versus $1/G$.

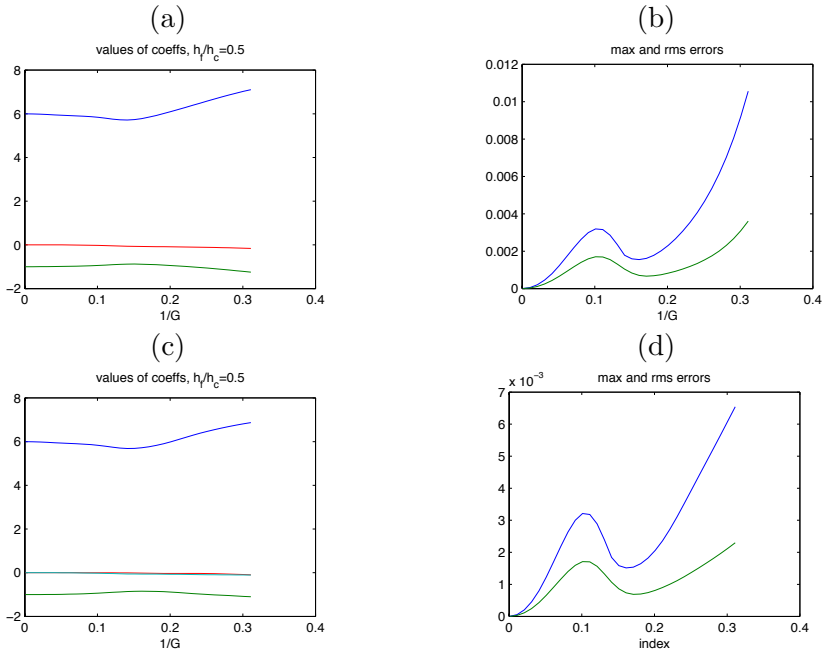


FIGURE 5. Optimized coefficients and phase speed errors in 3-D; (a), (b) the 15-point operator described in the text; (c), (d) the 27-point operator.

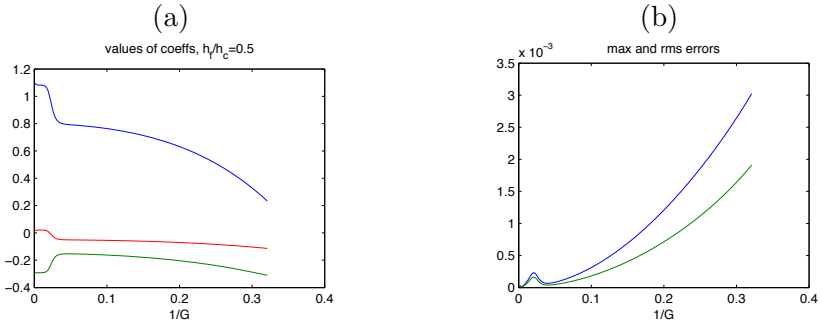


FIGURE 6. Optimized coefficients and phase speed errors for 2-D finite elements.

2.3. **Optimized finite difference scheme in 3-D.** In 3-D we similarly construct an optimized operator to match the standard seven point finite difference discretization of the Helmholtz operator. The coefficients of the optimized operator are again denoted by a_1, a_2, a_3, a_4. These coefficients correspond to contributions from the center, the face-centers, the edge-centers and the corners of a 27-point cube, such that the symbol becomes

$$\begin{aligned}\sigma^{\text{opt}} = & -k^2 + h^{-2}(a_1 + 2a_2(\cos(h\xi_1) + \cos(h\xi_2) + \cos(h\xi_3)) \\ & + 2a_3(\cos(h(\xi_1+\xi_2)) + \cos(h(\xi_1-\xi_2)) + \cos(h(\xi_1+\xi_3)) + \cos(h(\xi_1-\xi_3)) \\ & + \cos(h(\xi_2+\xi_3)) + \cos(h(\xi_2-\xi_3))) + 2a_4(\cos(h(\xi_1+\xi_2+\xi_3)) \\ & + \cos(h(\xi_1-\xi_2+\xi_3)) + \cos(h(\xi_1+\xi_2-\xi_3)) + \cos(h(\xi_1-\xi_2-\xi_3))).\end{aligned} \tag{17}$$

A very similar optimization method was applied, minimizing a weighted difference of σ^{opt} and σ^{fd7}. Two cases were studied, first the case with $a_3 = 0$, which leads to a 15-point operator, and then the case where all four coefficients are allowed to be nonzero. The values of the coefficients, and the rms and maximum errors in the phase speed are given in Figure 5.

2.4. **Optimized regular grid finite elements.** It is straightforward to derive the matrix for 2-D regular grid finite elements using first order rectangular (square) elements and constant k^2, see section 5.2 below. One finds

$$\begin{aligned}(H_h^{\text{FE}}u)_{i,j} = & (\frac{8}{3} - \frac{4}{9}h^2k^2)u_{i,j} + (-\frac{1}{3} - \frac{1}{9}h^2k^2)(u_{i+1,j} + u_{i-1,j} + u_{i,j+1} + u_{i,j-1}) \\ & + (-\frac{1}{3} - \frac{1}{36}h^2k^2)(u_{i+1,j+1} + u_{i-1,j+1} + u_{i+1,j-1} + u_{i-1,j-1})\end{aligned} \tag{18}$$

Note that H^{FE} scales differently with h due to the use of the finite element method instead of finite differences. It is straightforward to find optimized coarse scale finite difference operators in the form

$$\begin{aligned}H_h^{\text{opt,FE}}u(x_{i,j}) = & a_1u_{i,j} + a_2(u_{i-1,j} + u_{i+1,j} + u_{i,j-1} + u_{i,j+1}) \\ & + a_3(u_{i-1,j-1} + u_{i-1,j+1} + u_{i+1,j-1} + u_{i+1,j+1})\end{aligned} \tag{19}$$

The same optimization scheme is used, except that the points $(0,\pi), (\pi,\pi)$ are omitted from measure of symbol differences. The values of the coefficients, and the max and rms errors are given in Figure 6.

3. The multigrid methods

The above described operator is used in two-grid and multigrid methods. In this section we describe the multigrid methods used. For background on multigrid methods we refer to [16].

The multigrid method will be used as a *preconditioner* for a Krylov subspace method (GMRES [13]).

We next discuss the choice of smoother. Smoothers apply one or more steps of an iterative method for the equation

$$L_{\mathrm{S},h} u_h = f_h. \tag{20}$$

This generally involves a choice of the iterative method and of the number of smoothing steps (presmoothing and postsmoothing). Here the smoother will be denoted by S_h^ν, ν being the number of iterations, and will be based on ω-Jacobi or successive over relaxation (SOR) (which, when the relaxation factor is 1 of course reduces to Gauss Seidel (GS)).

However, we also consider the operator $L_{\mathrm{S},h}$ as subject to choice, setting it equal to a discretization of $-\Delta$ instead of the Helmholtz operator we are trying to invert. The reason to not choose $L_{\mathrm{S},h}$ equal to the Helmholtz equation has to do with the instability in the ω-Jacobi and SOR methods for the discrete Helmholtz operator, and will be further explained in section 4 on Fourier analysis of the multigrid methods below. Note that this type of modification is different from shifted Laplacian methods. In the latter both the operator used for smoothing and the operator used for coarse grid correction are modified to have additional damping, but they remain equal.

The choices of grid refinement, and of the restriction and prolongation operators are standard. We start with a square grid with grid size h, and apply standard coarsening to square grids with grid size $2h$, $4h$ etc. For the restriction operator we mostly use full weighting, i.e. it is the tensor product of one-dimensional FW operators of the form, in stencil notation, $\frac{1}{4}\begin{bmatrix}1 & 2 & 1\end{bmatrix}$. In 2-D it reads, in stencil notation

$$R_h = \frac{1}{16}\begin{bmatrix}1 & 2 & 1\\ 2 & 4 & 2\\ 1 & 2 & 1\end{bmatrix}, \tag{21}$$

or, if i, j are coordinates in the coarse grid

$$\begin{aligned}(R_h u)_{i,j} = \frac{1}{16}\Big(&4u_{2i,2j} + 2(u_{2i-1,2j} + u_{2i+1,2j} + u_{2i,2j-1} + u_{2i,2j+1})\\ &+ u_{2i-1,2j-1} + u_{2i+1,2j-1} + u_{2i-1,2j+1} + u_{2i+1,2j+1}\Big)\end{aligned} \tag{22}$$

As an alternative, the use of straight injection is briefly studied in section 4. The prolongation operator is given by

$$P_h = 4(R_h)'. \tag{23}$$

The two-grid approximate solution formula is then given by

$$M_h^{2h} = S_h^{\nu_2} K_h^{2h} S_h^{\nu_1}, \tag{24}$$

where

$$K_h^{2h} = I_h - P_h (L_{\mathrm{c},2h})^{-1} R_h L_h, \tag{25}$$

is the coarse grid correction operator. Here L_h and $L_{\mathrm{c},2h}$ denote discretizations of $-\Delta - k^2$, that are still to be specified. We focus particularly on the choice of coarse grid operator $L_{\mathrm{c},2h}$. For the two-grid method in 2-D the following pairs of fine and coarse grid operators will be considered

(1) Both using standard 5-point finite differences (fd5-fd5)
(2) Using standard 5-point finite differences at the fine level and a Galerkin approximation of this matrix at the coarse level (fd5-gal)
(3) Using standard 5-point finite differences at the fine level and the new optimized operator at the coarse level (fd5-opt)
(4) Using the operator from [9] at the fine and coarse levels (jss-jss)

For multigrid with more than two coarsening levels we consider only the optimized method where at each scale the optimized operators is chosen to approximate the fine level finite difference operator. This is done using the parameter $h_{\rm rel} = h_{\rm fine}/h_{\rm coarse}$, which assumes the values 1/2, 1/4, 1/8, etc. In this case the algorithm is based on the V-cycle.

Comparing the case of two levels with the case of more than two levels, the additional levels are added on the fine side, i.e. assuming identical coarsest grid. The coarse level behavior is expected to determine the behavior of the algorithm and therefore most emphasis is on the two-grid problem. A few experiments are done using multiple levels, to study whether it is indeed the two-grid performance determines the convergence behavior.

4. Fourier analysis of multigrid methods

In this section, we compute and analyze two-grid convergence factors by using local Fourier analysis as described in [16], chapters 3 and 4. We concentrate on the computation of the spectral radius $\rho(M_h^{2h})$ for the two-grid iteration operator M_h^{2h} that determines the asymptotic convergence behavior. We consider the 2-D case, using standard 5-point finite differences on the fine scale with an optimized finite difference scheme $L_{c,2h} = L_{2h}^{\rm opt}$ as well as the same five point finite difference operator $L_{c,2h} = L_{2h}^{\rm 5pt}$ for the coarse grid correction to show the efficiency of the optimized nine point finite difference scheme $L_{2h}^{\rm opt}$. We also study the case with the Jo-Shin-Suh operator on the fine and the coarse scale.

The analysis is done in the dimensionless Fourier domain, i.e. the wave vector is written as $h^{-1}\theta = h^{-1}(\theta_1, \theta_2)$, so that the dimensionless wave numbers θ are in $[-\pi, \pi)^2$ (in 2-D). Sets of "high" and "low" wave numbers are defined by

$$\begin{aligned} T^{\rm low} &= \Big[-\frac{\pi}{2}, \frac{\pi}{2}\Big)^2, \\ T^{\rm high} &= [-\pi, \pi)^2 \backslash T^{\rm low}. \end{aligned} \tag{26}$$

For $\theta = (\theta_1, \theta_2) \in T^{\rm low}$ we also define

$$\begin{aligned} \theta^{(0,0)} &= (\theta_1, \theta_2), & \theta^{(1,1)} &= (\bar\theta_1, \bar\theta_2). \\ \theta^{(0,1)} &= (\theta_1, \bar\theta_2), & \theta^{(1,0)} &= (\bar\theta_1, \theta_2), \end{aligned} \tag{27}$$

where

$$\bar\theta_i = \begin{cases} \theta_i + \pi & \text{if } \theta_i < 0 \\ \theta_i - \pi & \text{if } \theta_i \geq 0, \end{cases} \tag{28}$$

The two-grid operator was given in (24), (25). In local Fourier analysis, for each $\theta \in T^{\rm low}$ we define the four-dimensional space of harmonics by

$$E^\theta = \text{span}\left(\left\{ e^{i\theta^\alpha \cdot x/h} \,|\, \alpha \in \{(0,0),(1,1),(0,1),(1,0)\} \right\}\right) \tag{29}$$

The two-grid operator maps this space into itself, and hence can be reduced to a 4×4 matrix for each $\theta \in T^{\rm low}$, which will be denoted by $\hat M_h^{2h}(\theta)$. It can be written as

$$\hat M_h^{2h}(\theta) = \hat S_h(\theta)^{\nu_2} \hat K_h^{2h}(\theta) \hat S_h(\theta)^{\nu_1}, \qquad \theta \in T^{\rm low}, \tag{30}$$

where

$$\hat K_h^{2h}(\theta) = \hat I_h - \hat P_h(\theta)(\hat L_{c,2h}(2\theta))^{-1} \hat R_h(\theta) \hat L_h(\theta), \tag{31}$$

where $\hat I_h$ is the 4×4 identity matrix and the meaning of $\hat K_h^{2h}(\theta)$, $\hat P_h(\theta)$, $\hat L_{c,2h}(2\theta)$, $\hat R_h(\theta)$ and $\hat L_h(\theta)$ will be explained next. The Fourier transformed restriction and prolongation operators $\hat R_h$ and $\hat P_h$ are 1×4 and 4×1 matrices respectively, given by

$$\hat R_h = \begin{pmatrix} \frac{1}{4}(1+\cos\theta_1)(1+\cos\theta_2) & \frac{1}{4}(1+\cos\bar\theta_1)(1+\cos\bar\theta_2) & \frac{1}{4}(1+\cos\theta_1)(1+\cos\bar\theta_2) & \frac{1}{4}(1+\cos\bar\theta_1)(1+\cos\theta_2) \end{pmatrix} \tag{32}$$

and

$$\hat{P}_h = \begin{pmatrix} \frac{1}{4}(1+\cos\theta_1)(1+\cos\theta_2) \\ \frac{1}{4}(1+\cos\bar{\theta}_1)(1+\cos\bar{\theta}_2) \\ \frac{1}{4}(1+\cos\bar{\theta}_1)(1+\cos\theta_2) \\ \frac{1}{4}(1+\cos\theta_1)(1+\cos\bar{\theta}_2), \end{pmatrix} \tag{33}$$

The operator $\hat{L}_{c,2h}(2\theta)$ is a scalar given by the value of the symbol $\sigma(\theta^{(0,0)}/h)$. The operator $\hat{L}_h$ is a diagonal 4×4 matrix given by

$$\hat{L}_h(\theta) = \mathrm{diag}(\tilde{L}_h(\theta^{(0,0)}), \tilde{L}_h(\theta^{(1,1)}), \tilde{L}_h(\theta^{(0,1)}), \tilde{L}_h(\theta^{(1,0)})\,), \tag{34}$$

in which $\tilde{L}_h(\theta)$ is the scalar value of the symbol $\sigma(\theta/h)$. The matrix $\hat{S}_h(\theta)$ is also a 4×4 diagonal matrix

$$\hat{S}_h(\theta) = \mathrm{diag}(\tilde{S}_h(\theta^{(0,0)}), \tilde{S}_h(\theta^{(1,1)}), \tilde{S}_h(\theta^{(0,1)}), \tilde{S}_h(\theta^{(1,0)})\,), \tag{35}$$

containing the *smoothing factors* $\tilde{S}_h(\theta)$, which we now discuss.

The smoothing factors depend on the iterative method and choice of discretization, see [16] for their derivation. For standard 5-point finite differences of the Helmholtz operator and ω-Jacobi they are given by

$$\tilde{S}_h(\theta) = 1-\omega+\frac{2\omega}{4-k^2h^2}(\cos\theta_1+\cos\theta_2). \tag{36}$$

and for SOR by

$$\tilde{S}_h(\theta) = \frac{(1-\omega)(4-k^2h^2)+\omega(\exp(i\theta_1)+\exp(i\theta_2))}{4-k^2h^2-\omega(\exp(-i\theta_1)+\exp(-i\theta_2))}. \tag{37}$$

This concludes the explanation of the computation of $\hat{M}_h^{2h}$. Note that $\rho_{\mathrm{loc}}(M_h^{2h})$ is given by

$$\rho_{\mathrm{loc}}(M_h^{2h}) = \sup\left\{\rho_{\mathrm{loc}}(\hat{M}_h^{2h}(\theta))\,|\,\theta\in T^{\mathrm{low}}\right\}, \tag{38}$$

and that $\rho_{\mathrm{loc}}(\hat{M}_h^{2h}(\theta))$ is the spectral radius of the 4×4 matrix $\hat{M}_h^{2h}(\theta)$.

4.1. **Using the Laplacian for smoothing.** Using the smoothing factors just computed, the choice of using the Laplacian for smoothing can be motivated. From (36) it follows that, for $\theta=(\theta_1,\theta_2)=(0,0)$

$$\tilde{S}_h(\theta) = 1-\omega+\frac{4\omega}{4-k^2h^2} > 1 \tag{39}$$

and hence the iterative smoothing method is unstable in this case. If $kh=0$ on the other hand the smoothing factor satisfies

$$1-2\omega \le \tilde{S}_h(\theta) \le 1. \tag{40}$$

Similarly observations can be made for (37). Hence using a discretization of the operator $-\Delta$ in the smoothing steps means the instability disappears (no blow up of errors for $\theta\approx(0,0)$). However, this use of a different operator means that the smoothing method doesn't converge to the true solution. Instead, in the normalized Fourier domain, it converges to

$$\frac{4-2\cos(\theta_1)-2\cos(\theta_2)-k^2h^2}{4-2\cos(\theta_1)-2\cos(\theta_2)} = 1-\frac{k^2h^2}{4-2\cos(\theta_1)-2\cos(\theta_2)} \tag{41}$$

times the solution. The purpose of the smoother is to invert the equation for the high wave numbers, as for the low wave numbers the coarse grid correction is used. For $(\theta_1,\theta_2)\in T_h^{\mathrm{high}}$ the factor (41) is somewhere between 0.21 and 1 assuming that $G_c\ge 4$ and hence $kh\le\frac{2\pi}{4}$. Typically this kind of factor is easily dealt with by the outer Krylov method, and we conjecture that the stabilizing effect of our choice of $L_{S,h}$ outweighs the error

α	0.0025			0.005			0.01		
(ν_1, ν_2)	$(1,1)$	$(1,2)$	$(2,2)$	$(1,1)$	$(1,2)$	$(2,2)$	$(1,1)$	$(1,2)$	$(2,2)$
SOR1.2	> 1	0.4710	0.2320	0.7956	0.2816	0.2320	0.5024	0.2320	0.2320
SOR1.1	> 1	0.3002	0.2344	0.6646	0.2320	0.2320	0.4067	0.2320	0.2320
GS	> 1	0.3244	0.2569	0.6882	0.2320	0.2320	0.4064	0.2320	0.2320
SOR0.9	> 1	0.5014	0.2871	0.9235	0.2966	0.2320	0.5356	0.2330	0.2320
SOR0.8	> 1	0.9096	0.3513	> 1	0.5249	0.2327	0.7678	0.3083	0.2320
SOR0.7	> 1	> 1	0.7272	> 1	0.8927	0.4224	> 1	0.5178	0.2727
J	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
0.9J	> 1	0.5120	0.4096	0.8373	0.5120	0.4096	0.6400	0.5120	0.4096
0.8J	> 1	0.9487	0.4054	> 1	0.5599	0.2867	0.7723	0.3607	0.2419
0.7J	> 1	> 1	0.8041	> 1	0.9418	0.4776	> 1	0.5563	0.3253
0.6J	> 1	> 1	> 1	> 1	> 1	0.8368	> 1	0.8434	0.4969
0.5J	> 1	> 1	> 1	> 1	> 1	> 1	> 1	> 1	0.7927

TABLE 1. Two-grid convergence factors $\rho_{\text{loc}}(M_h^{2h})$, $G = 4$, fd5-opt.

introduced. This is confirmed in the numerical simulations below. For the Gauss-Seidel method similar behavior occurs.

4.2. **Computation of two-grid convergence factors.** The two-grid convergence factors are computed using the expression (38), taking the supremum for θ on regular grid. This grid was restricted to the first quadrant because of the symmetry, and taken large enough to accurately compute the maximum in (38). We have computed the convergence factor for three schemes

(1) The fd5-opt scheme using the five point scheme in the fine grid along with optimized scheme for the coarse grid
(2) The fd5-fd5 scheme using the five point scheme on both the fine and the coarse grids
(3) The jss-jss scheme using the JSS operator on both the fine and the coarse grids.

In Table 1 we present $\rho_{\text{loc}}(M_h^{2h})$ and for the fd5-opt scheme, using SOR as smoother with $\omega = 0.8, 0.9, 1.0$ and 1.1, and with ω-Jacobi as a smoother, with $\omega = 0.7, 0.8, 0.9$ and 1.0. In this case the performance for $G_c = 4$ studied. In Table 2 results for the fd5-fd5 method are given, using the same smoothing methods and $G_c = 8$. Best performance occurs roughly for SOR with $\omega = 1$ (i.e. the Gauss-Seidel method) and for ω-Jacobi with $\omega = 0.8$ or 0.9. It is clearly observed that the fd5-opt method requires smaller α for convergence, and that it has good convergence factors for $G_c = 4$. For the fd5-fd5 method $G_c = 8$ was taken to establish a good choice of smoothing method, as lower values of G_c gave poor convergence. We have also investigated the SI operator for restriction. However, the performance was very poor compared to the full weighting restriction operator.

In Table 3 we vary both α and the number of grid points per wave length in the coarse grid. In this table we also included results for the jss-jss method. We observe that the fd5-opt and jss-jss methods are comparable, and much better than fd5-fd5. Concerning the number of smoothing steps, for GS $(\nu_1, \nu_2) = (1,1)$ appears a good choice and $(\nu_1, \nu_2) = (2,2)$ for ω-Jacobi. Clearly the damping α plays an important role in the convergence.

α	0.005			0.01			0.02		
(ν_1, ν_2)	$(1,1)$	$(1,2)$	$(2,2)$	$(1,1)$	$(1,2)$	$(2,2)$	$(1,1)$	$(1,2)$	$(2,2)$
SOR1.1	> 1	> 1	> 1	> 1	> 1	> 1	0.8128	0.7329	0.6661
GS	> 1	> 1	> 1	> 1	> 1	> 1	0.8151	0.7660	0.7091
SOR0.9	> 1	> 1	> 1	> 1	> 1	> 1	0.8364	0.7932	0.7469
SOR0.8	> 1	> 1	> 1	> 1	> 1	> 1	0.8893	0.8221	0.7807
J	> 1	> 1	> 1	> 1	> 1	> 1	1.0000	1.0000	1.0000
0.9J	> 1	> 1	> 1	> 1	> 1	> 1	0.9135	0.8668	0.8338
0.8J	> 1	> 1	> 1	> 1	> 1	> 1	0.9568	0.8864	0.8498
0.7J	> 1	> 1	> 1	> 1	> 1	> 1	> 1	0.9193	0.8716

TABLE 2. Two-grid convergence factors $\rho_{\text{loc}}(M_h^{2h})$ for the fd5-fd5 method and $G_{\text{c}} = 8$.

	$\alpha = 0.00125$			$\alpha = 0.005$			$\alpha = 0.02$		
$G = 3.5$	fd5-opt	fd5-fd5	jss-jss	fd5-opt	fd5-fd5	jss-jss	fd5-opt	fd5-fd5	jss-jss
GS	> 1		> 1	0.9147		> 1	0.3043		0.3990
0.8J	0.9646		> 1	0.3264		> 1	0.2901		0.2986
0.9J	0.5722		> 1	0.4096		0.9307	0.4096		0.4096
$G = 4$									
GS	> 1		> 1	0.6865		0.7666	0.2794		0.3019
0.8J	0.4054		> 1	0.2866		0.5265	0.2320		0.1572
0.9J	0.4096		> 1	0.4096		0.4096	0.4096		0.4096
$G = 5$									
GS	> 1		> 1	0.4224		0.5151	0.2302		0.2029
0.8J	0.6439		0.8030	0.2591		0.2576	0.1811		0.1388
0.9J	0.6148		0.7522	0.4096		0.4096	0.4096		0.4096
$G = 6$									
GS	0.7605	> 1	> 1	0.3044	> 1	0.3891	0.1881	> 1	0.1613
0.8J	0.7612	> 1	0.9803	0.2577	> 1	0.3109	0.1561	> 1	0.1303
0.9J	0.7454	> 1	> 1	0.4096	> 1	0.4096	0.4096	> 1	0.4096
$G = 7$									
GS	> 1	> 1	> 1	0.3034	> 1	0.3446	0.1797	1.0058	0.1613
0.8J	> 1	> 1	> 1	0.3131	> 1	0.3047	0.1468	1.0585	0.1296
0.9J	> 1	> 1	> 1	0.4096	> 1	0.4096	0.4096	1.0316	0.4096
$G = 8$									
GS	> 1	> 1	> 1	0.3864	> 1	0.2803	0.1840	0.8126	0.1614
0.8J	> 1	> 1	> 1	0.3948	> 1	0.2819	0.1472	0.8456	0.1296
0.9J	> 1	> 1	> 1	0.4096	> 1	0.4096	0.4096	0.8299	0.4096

TABLE 3. Two-grid convergence factors as a function of α, G, choice of smoothing method for fd5-opt, fd5-fd5 and jss-jss.

5. Numerical results and discussion

In this section we compare the convergence of multigrid methods with different choices of the coarse scale operators. We start with several experiments for the two-grid method in 2-D. Then we investigate the extension to the multigrid method and to the 3-D case, in order to establish that the behavior observed in the 2-D, two-grid experiments also occurs more generally.

5.1. **The two-grid method in 2-D.** For a full comparison we study the trade-off between damping present in the equation, the number of grid points per wave length at the coarse level G_{c} and the number of iterations required to reduce the residual by a factor of 10^{-6}.

In 2-D it is relatively easy to perform this kind of experiments, as in 3-D the size of the coarse scale problem quickly becomes large for modern desktop machine (a machine with 8GB memory was used for these experiments using a Matlab implementation).

We now specify the methods use in these simulations. The regular five point operator at the fine scale is combined at the coarse scale with three choices of coarse scale operator, namely the regular five point operator, the Galerkin operator and our optimized method. These combinations are denoted fd5-fd5, fd5-gal and fd5-opt. We also study the JSS operator, used both at the fine and at the coarse scale, referred to as jss-jss. As shown below, sharp differences are present between the optimized coarse scale methods on the one hand and the fd5 and Galerkin coarse scale methods on the other hand.

The first series of experiments concerned a constant coefficient medium, i.e. k is constant equal to $\frac{\pi}{G_c h}$, where G_c is the number of grid points per wavelength at the coarse scale. Experiments were performed with three variants for the smoothing method, first the Gauss-Seidel (GS) method, secondly Jacobi relaxation with $\omega = 0.8$ (J0.8) and thirdly the Jacobi relaxation with $\omega = 0.9$. In the first case the number of pre- and postsmoothing steps equalled $(n_1, n_2) = (1, 1)$, in the second and third case the choice $(n_1, n_2) = (2, 2)$ was used. We found that all three methods performed comparably for the 5-point and Galerkin coarse scale operators. For the JSS method the second and third method performed comparably and better than the first. For the opt method the third method performed mostly better than the second, with very few exceptions, and both performed better than GS. We therefore display in these experiments only the results for the Jacobi relaxation with $\omega = 0.9$ and $(n_1, n_2) = (2, 2)$.

The results are given in Table 4. The following can be clearly observed

- The fd5-fd5 and fd5-gal method require relatively large G_c, say ≥ 7 as well as large damping of $\alpha = 0.01$ or 0.02 or larger.
- The fd5-opt and jss-jss methods perform well with G_c as low as 3.5 and $\alpha = 0.0025$ or $\alpha = 0.005$ or larger.

Here $N_{\text{iter}} \lesssim 20$ is used as the (somewhat subjective) criterion for good convergence. Clearly the optimized methods (jss and opt) perform *much* better than the conventional choices fd5 or gal at the coarse level.

Next we test this in two variable coefficient media. The first is a random medium, for which results are given in Table 5, and the second is the Marmousi model, see Table 6. The Marmousi model has size 9200 x 3000 meters, and wave speeds between 1500 and 5500 ms^{-1}, see the plot. The value of G_c is the minimum value present in the model. For the random medium, this minimum is assumed only in small, isolated regions of the model, for the Marmousi model it is assumed in the top-layer, i.e. a larger region. This perhaps explains the small differences, with the convergence in the random medium starting roughly at $G_c = 3$ for the fd5-opt and jss-jss methods.

5.2. **Further experiments: Multigrid, 3-D and PML layers.** Further numerical experiments are done to establish whether these result extend to multigrid experiments, to 3-D and to examples with PML layers. Our method to include PML layers is new and is described in this section.

Results for 2-D multigrid with 2, 3 and 4 levels, using a constant coefficient medium, are given in Table 7. In each case the same coarse level grid is used. For small G, the convergence deteriorates, but from $G_c \gtrsim 4$ the multigrid method converges about as fast as the two-grid method. For the random medium in Table 5, similar behavior is observed, but in this example the multigrid method converges about as fast as the two-grid method from $G_c \gtrsim 3.5$.

In 3-D the cost and memory use of the sparse factorizations scales worse than in 2-D. In this paper we study only a relatively small example of size 80^3 that can be done on a

G_c	N_{iter} for fd5-fd5,nx=511					N_{iter} for fd5-gal,nx=511				
	α=1.25e-3	2.5e-3	0.005	0.01	0.02	α=1.25e-3	2.5e-3	0.005	0.01	0.02
6	>100	>100	>100	94	33	>100	>100	>100	74	27
7	>100	>100	>100	55	22	>100	>100	>100	46	20
8	>100	>100	>100	36	16	>100	>100	93	33	15
10	>100	>100	49	21	11	>100	>100	45	20	11
12	>100	71	28	14	9	>100	60	26	14	9
G_c	N_{iter} for fd5-opt,nx=1023					N_{iter} for jss-jss,nx=1023				
	α=1.25e-3	2.5e-3	0.005	0.01	0.02	α=1.25e-3	2.5e-3	0.005	0.01	0.02
3	>100	>100	>100	>100	>100	>100	>100	>100	56	26
3.5	93	33	19	15	13	>100	97	35	18	13
4	43	20	13	10	9	50	21	13	10	9
5	22	12	9	7	7	26	13	9	7	7
6	23	12	8	7	6	37	15	9	7	6
7	44	17	10	7	6	35	15	9	7	6
8	89	25	12	8	6	26	13	8	6	5
10	94	25	12	8	7	18	10	7	6	5
12	34	16	10	7	6	12	8	6	5	4

TABLE 4. Convergence results for the two-grid method in a constant medium using different combinations of fine and coarse scale operators.

regular machine with 8 GB memory using Matlab. This was approximately the largest example that could be done in this setup. Two types of optimized operators where used, a 15-point discretization, using points in the center, on the faces and the corners of a 27-point cube, and a 27-point discretization using all points in a 27-point cube. The factorization of the 15-point operator used about 30 % less memory than that of the 27-point operator. The results on the unit cube with constant wave speed are given in Table 8. We use only our optimized method. Generalization of the method of [9] to 3-D exist [11, 3] but we have not tested these.

In 3-D, the observed convergence behavior as a function of α and G_c does not differ much from the behavior observed in 2-D.

We next consider the 2-D problem with PML boundary layers. This is important in practice – rectangular domains are often encountered in combination with PML boundary layers, for example in the seismic problem [11]. As mentioned, we consider a finite element discretization for this problem.

As explained in [10], in the PML method for the boundary $x_1 =$ constant the derivative $\frac{\partial}{\partial x_1}$ is replaced by $\frac{1}{1+i\omega^{-1}\sigma_1(x_1)}\frac{\partial}{\partial x_1}$. Here σ_1 is chosen to be 0 on the internal domain (no damping), and increases quadratically from the onset of the damping layer to boundary of the computational domain. We introduce PML layers on all four boundaries. It is convenient to multiply the equation by $\big(1+i\omega^{-1}\sigma_1(x_1)\big)\big(1+i\omega^{-1}\sigma_2(x_2)\big)$, this leads to a symmetric operator easily discretized by finite elements.

Straightforward inclusion of PML layers in the finite difference multigrid method was observed to lead to very poor convergence, or no convergence at all. Therefore we consider a coarsening strategy where inside the PML layer no coarsening takes place in the direction normal to the boundary. The resulting grids are given (schematically) in Figure 7. We consider first order rectangular elements on these grids, denoted by ϕ_{ij}. The matrix

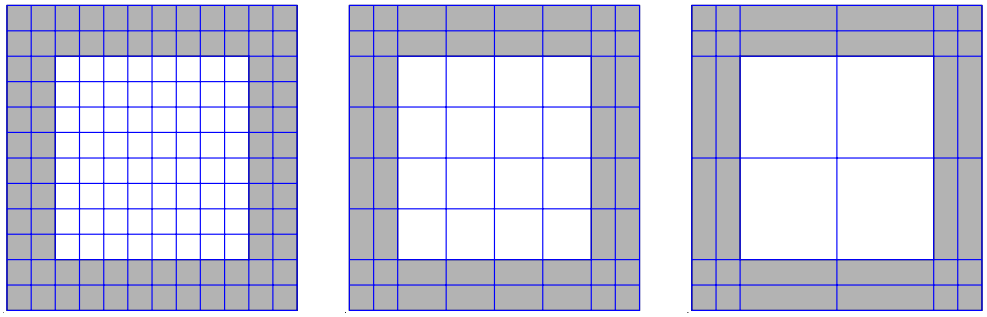


FIGURE 7. Coarsening strategy to handle PML layers. Inside the PML layers (in grey) there is no coarsening in the direction normal to the boundary.

elements are given by

$$M_{ij;kl} = \int \left[\alpha_1 \alpha_2^{-1} \frac{\partial \phi_{ij}}{\partial x_1} \frac{\partial \phi_{kl}}{\partial x_1} + \alpha_1^{-1} \alpha_2 \frac{\partial \phi_{ij}}{\partial x_2} \frac{\partial \phi_{ij}}{\partial x_2} - k(x)^2 \alpha_1^{-1} \alpha_2^{-1} \phi_{kl}(x) \phi_{kl} \right] dx, \tag{42}$$

where the function $\alpha_j(x_j) = \frac{1}{1+i\omega^{-1}\sigma_j(x_j)}$ contains the modifications for inclusion of the PML layer.

In the finite element context, the prolongation operator P follows straightforwardly from specifying the grids, while the restriction operator R is given by P'. Based on experiments with constant coefficient example we choose as a smoother one iteration of the ω-Jacobi method for the finite element discretization of $-\Delta + k^2$, with relaxation factor $\omega = 0.8$.

For the coarse grid operators the optimized stencils where computed in section 2. These are used in the internal, square grid part of the domain, in such a way that the corresponding rows of the matrix contain the optimized stencil coefficients. For the other rows the regular finite element matrix elements are used. We expect that the fact that no optimized coefficients are used inside the PML layer is of little importance, because the accuracy of the phase speeds is most relevant for long distance wave propagation, and there is no such propagation in the PML layer due to the damping.

With this method a number of numerical experiments was carried out, varying several parameters. Two choices of medium were considered, the constant and the random medium, both on the unit square with PML layers added outside it. A point source at $(0.3, 0.3)$ was used a right hand side. We experimented with different multigrid levels from 2 to 4, using nx $\times$ ny= 200 $\times$ 200 grid points at the coarse level, and with different sizes for the twogrid method, using nx = 200, 400 and 800. Each of the cases was carried out with $G_c = 4$ and $G_c = 6$ gridpoints per wavelength at the coarse level. The results are displayed in Table 9. We see that the number of iterations is mostly low (≤ 20 iterations) and increases above this only for the experiments with 3 and 4 levels with a constant medium and $G = 4$ (the most difficult case).

Hence the new coarse grid operators can be used with PML boundary layers as well, providing the same kind of reduction in G_c as for the examples without PML.

REFERENCES

[1] J.-P. Berenger. A perfectly matched layer for the absorption of electromagnetic waves. *Journal of Computational Physics*, 114(2):185–200, 1994.

[2] M. Bollhöfer, M. J. Grote, and O. Schenk. Algebraic multilevel preconditioner for the Helmholtz equation in heterogeneous media. *SIAM J. Sci. Comput.*, 31(5):3781–3805, 2009.

[3] Z. Chen, D. Cheng, and T. Wu. A dispersion minimizing finite difference scheme and preconditioned solver for the 3D Helmholtz equation. *Journal of Computational Physics*, 231(24):8152–8175, 2012.

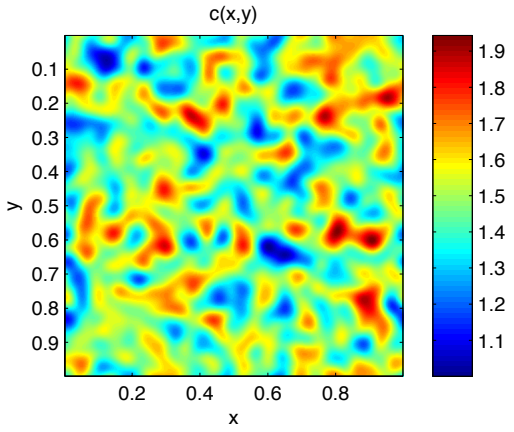
[4] W. C. Chew and W. H. Weedon. A 3D perfectly matched medium from modified Maxwell's equations with stretched coordinates. *Microwave and optical technology letters*, 7(13):599–604, 1994.
[5] H. C. Elman, O. G. Ernst, and D. P. O'Leary. A multigrid method enhanced by Krylov subspace iteration for discrete Helmhotz equations. *SIAM J. Sci. Comput.*, 23(4):1291–1315 (electronic), 2001.
[6] B. Engquist and L. Ying. Sweeping preconditioner for the Helmholtz equation: moving perfectly matched layers. *Multiscale Model. Simul.*, 9(2):686–710, 2011.
[7] Y. A. Erlangga, C. W. Oosterlee, and C. Vuik. A novel multigrid based preconditioner for heterogeneous Helmholtz problems. *SIAM J. Sci. Comput.*, 27(4):1471–1492 (electronic), 2006.
[8] A. George. Nested dissection of a regular finite element mesh. *SIAM J. Numer. Anal.*, 10:345–363, 1973.
[9] Jo, Churl-Hyun and Shin, Changsoo and Suh, Jung Hee. An optimal 9-point, finite-difference, frequency-space, 2-D scalar wave extrapolator. *Geophysics*, 61(2):529–537, 1996.
[10] S. G. Johnson. Notes on perfectly matched layers. http://math.mit.edu/ stevenj/18.369/pml.pdf, 2010.
[11] S. Operto, J. Virieux, P. Amestoy, J.-Y. L'Excellent, L. Giraud, and H. B. H. Ali. 3D finite-difference frequency-domain modeling of visco-acoustic wave propagation using a massively parallel direct solver: A feasibility study. *Geophysics*, 72(5, S):SM195–SM211, 2007.
[12] J. Poulson, B. Engquist, S. Fomel, S. Li, and L. Ying. A parallel sweeping preconditioner for high-frequency heterogeneous 3d helmholtz equations. Technical report, University of Texas at Austin, 2012. arxiv.org/1204.0111.
[13] Y. Saad and M. H. Schultz. GMRES: a generalized minimal residual algorithm for solving nonsymmetric linear systems. *SIAM J. Sci. Statist. Comput.*, 7(3):856–869, 1986.
[14] C. C. Stolk. A rapidly converging domain decomposition method for the helmholtz equation. *Journal of Computational Physics*, 241:240 – 252, 2013.
[15] L. N. Trefethen. Group velocity in finite difference schemes. *SIAM Rev.*, 24(2):113–136, 1982.
[16] U. Trottenberg, C. W. Oosterlee, and A. Schüller. *Multigrid*. Academic Press Inc., San Diego, CA, 2001. With contributions by A. Brandt, P. Oswald and K. Stüben.
[17] S. Wang, M. V. De Hoop, and J. Xia. On 3D modeling of seismic wave propagation via a structured parallel multifrontal direct Helmholtz solver. *Geophysical Prospecting*, 59:857–873, 2011.

Korteweg-de Vries Institute for Mathematics, University of Amsterdam, Science Park 904, 1098 XH Amsterdam

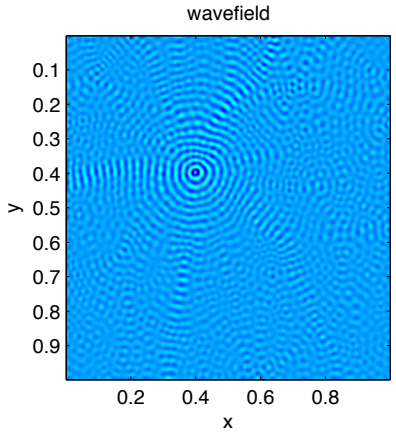
Department of Mathematics, Jagannath University, Dhaka-1100, Bangladesh

Department of Mathematics, University of Dhaka, Dhaka-1000, Bangladesh

(a) velocity



(b) solution with nx $= 511, G = 4, \alpha =$ 2.5e-3

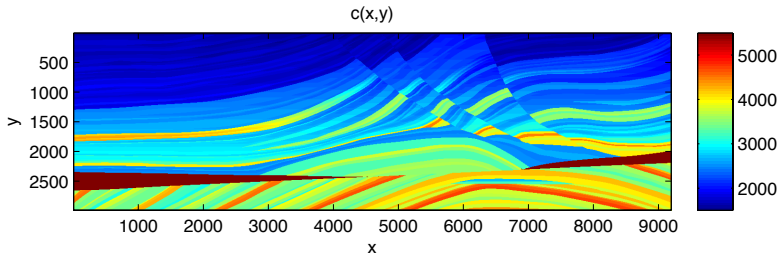


G_c	N_{iter} for fd5-fd5,nx=511					N_{iter} for fd5-gal,nx=511				
	α=1.25e-3	2.5e-3	0.005	0.01	0.02	α=1.25e-3	2.5e-3	0.005	0.01	0.02
5	>100	>100	>100	40	18	>100	>100	97	35	17
6	>100	>100	60	24	13	>100	>100	54	22	12
7	>100	>100	37	17	10	>100	>100	35	17	10
8	>100	70	26	14	9	>100	66	25	13	9
10	88	33	16	10	7	88	33	16	10	7
12	41	21	12	8	7	44	21	12	8	7

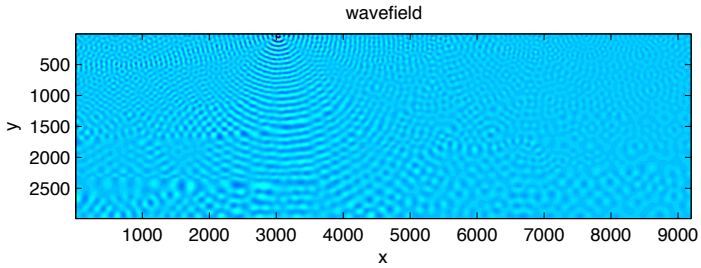
G_c	N_{iter} for fd5-opt,nx=1023					N_{iter} for jss-jss,nx=1023				
	α=1.25e-3	2.5e-3	0.005	0.01	0.02	α=1.25e-3	2.5e-3	0.005	0.01	0.02
2.5	>100	>100	64	48	39	>100	>100	49	27	18
3	22	15	12	11	10	27	17	13	11	10
3.5	14	11	9	8	8	12	9	8	8	8
4	12	9	8	7	7	12	9	8	7	7
5	14	10	8	7	6	11	9	7	6	6
6	16	11	8	7	6	10	8	6	6	5
7	14	10	8	7	6	9	7	6	5	5
8	12	9	7	7	6	8	6	5	5	4

TABLE 5. Convergence results for different methods for a random medium

(a) medium



(b) solution for nx $= 1149, \alpha =$ 2.5e-3, $G_c = 4$



G_c	N_{iter} for fd5-fd5,nx=2299					N_{iter} for fd5-gal,nx=2299				
	α=1.25e-3	2.5e-3	0.005	0.01	0.02	α=1.25e-3	2.5e-3	0.005	0.01	0.02
5	>100	>100	>100	>100	39	>100	>100	>100	78	31
6	>100	>100	>100	51	23	>100	>100	>100	44	20
7	>100	>100	70	30	16	>100	>100	63	27	15
8	>100	>100	44	21	12	>100	>100	41	20	12
10	>100	52	24	14	9	>100	50	23	13	9
12	66	28	16	10	8	67	28	15	10	8

G_c	N_{iter} for fd5-opt,nx=2299					N_{iter} for jss-jss,nx=2299				
	α=1.25e-3	2.5e-3	0.005	0.01	0.02	α=1.25e-3	2.5e-3	0.005	0.01	0.02
2.5	>100	>100	>100	>100	>100	>100	>100	>100	>100	95
3	>100	>100	>100	>100	77	>100	>100	92	38	20
3.5	43	28	20	15	12	>100	53	27	16	11
4	33	21	15	11	8	35	23	16	11	9
5	24	16	12	9	7	22	16	12	9	7
6	25	15	10	8	7	17	12	10	8	6
7	27	15	10	8	7	15	11	9	7	6
8	26	16	10	8	7	13	10	8	7	6

TABLE 6. Convergence results for different methods for the Marmousi example

	N_{iter} for fd5-opt, nxcoarse=255								
	α =2.5e-3			α =0.005			α =0.01		
G_c	nlevels=2	3	4	nlevels=2	3	4	nlevels=2	3	4
3	>100	>100	>100	>100	>100	>100	>100	>100	>100
3.5	32	39	>100	19	28	>100	15	26	>100
4	20	18	20	13	13	14	10	12	13
5	12	13	13	9	9	10	7	8	8
6	12	13	14	8	9	9	7	7	7
7	15	20	22	9	11	12	7	8	8
8	19	29	32	12	14	15	8	9	9
	N_{iter} for jss-jss, nxcoarse=255								
	α =2.5e-3			α =0.005			α =0.01		
G_c	nlevels=2	3	4	nlevels=2	3	4	nlevels=2	3	4
3	>100	>100	>100	>100	>100	>100	55	66	>100
3.5	94	77	70	34	31	29	18	18	18
4	21	15	17	13	12	12	10	10	11
5	13	21	25	9	12	13	7	9	10
6	15	22	24	9	12	12	7	8	9
7	13	19	20	9	11	11	7	8	8
8	11	14	15	8	9	10	6	7	7

TABLE 7. Multigrid convergence for the constant medium with 2,3 and 4 levels, and different values of G and α

G_c	N_{iter} for fd7-opt, 15pt opt method					N_{iter} for fd7-opt, 27pt opt method				
	α=1.25e-3	2.5e-3	0.005	0.01	0.02	α=1.25e-3	2.5e-3	0.005	0.01	0.02
3	>100	>100	>100	>100	>100	>100	>100	>100	>100	>100
3.5	>100	65	30	21	18	>100	45	26	20	18
4	28	22	16	12	11	26	20	15	12	10
5	14	12	9	8	7	14	12	9	8	7
6	10	9	7	7	6	10	9	7	7	6
7	8	8	7	6	6	8	8	7	6	6
8	8	8	7	6	6	8	8	7	6	6
10	6	6	6	6	5	6	6	6	6	5
12	5	5	5	5	5	5	5	5	5	5

TABLE 8. Two-grid convergence for a constant coefficient medium in 3-D using 15 point and 27 point optimized operators at the coarse level.

	constant medium $G = 4$	constant medium $G = 6$	random medium $G = 4$	random medium $G = 6$
nlevels=2, nxc=200	16	12	11	10
nlevels=3, nxc=200	23	16	16	13
nlevels=4, nxc=200	35	20	20	15
nlevels=2, nxc=200	16	12	11	10
nlevels=2, nxc=400	17	12	12	11
nlevels=2, nxc=800	20	13	13	13

TABLE 9. Iteration numbers for the numerical experiments with PML layer.